**Client Requirements Specification**

**Project Overview**

This document outlines the requirements for the development of a mobile and web application that provides home product repair services. The application will allow customers to find repair services for products such as ACs, TVs, fans, fridges, etc. When a customer needs a service, they will raise a service request (referred to as a token) through the app, which will then be forwarded to the appropriate mechanic using a different application.

**1. Objective**

The goal of this project is to develop a user-friendly mobile and web application that connects customers needing home product repairs with mechanics who can provide these services. The system will streamline the process of requesting, assigning, and completing repairs.

**2. Scope**

* **Platform:** The application will be available on Android, iOS, and as a web application.
* **Users:**
  + **Customers:** End-users who require repair services for home products.
  + **Mechanics:** Service providers who will receive and respond to service requests.

**3. Key Features**

**3.1. Customer Mobile and Web App**

* **User Registration & Login:**
  + Customers should be able to sign up and log in using email, phone number, or social media accounts.
* **Service Listing:**
  + A list of available repair services for home products (e.g., AC, TV, fan, fridge).
  + Search and filter options for services.
* **Service Request (Token) Generation:**
  + Option for customers to select a service and raise a request (referred to as a token).
  + The token should include details such as the product type, issue description, and preferred service date/time.
* **Service Tracking:**
  + Customers should be able to track the status of their service request in real time.
* **Payment Gateway:**
  + Integration with payment gateways to facilitate online payments for services.
* **Notification System:**
  + Push notifications, SMS, or email alerts for service request updates.
* **Customer Support:**
  + An integrated chat or support system to help customers with their queries.

**3.2. Mechanic Mobile and Web App**

* **User Registration & Login:**
  + Mechanics should be able to sign up and log in using email or phone number.
* **Service Request Management:**
  + Mechanics will receive tokens generated by customers.
  + The app should provide an interface to view, accept, or decline service requests.
* **Service Execution Tracking:**
  + Mechanics should be able to update the status of the service request (e.g., In Progress, Completed).
* **Earnings Dashboard:**
  + Mechanics should have access to a dashboard that tracks their earnings and completed services.
* **Notification System:**
  + Push notifications, SMS, or email alerts for new service requests and updates.

**4. Functional Requirements**

**4.1. Customer Mobile and Web App**

* **FR1: User Registration & Login**
  + The system should allow customers to sign up and log in using their email, phone number, or social media accounts.
  + The system should verify the user's email or phone number during registration.
* **FR2: Company Information Page**
  + The app should provide a dedicated page containing information about the company, including its mission, services offered, contact details, and terms of service.
* **FR3: Service Listing**
  + The system should allow customers to browse and search for available repair services by category (e.g., AC, TV, fan, fridge).
  + The system should enable filtering services based on product type, issue severity, and location.
* **FR4: Service Request (Token) Generation**
  + The system should allow customers to create and submit a service request (token) by selecting a service, providing a description of the issue, and specifying the preferred date and time for the service.
  + The token should be uniquely identified and include all relevant details for the mechanic.
* **FR5: Live Tracking**
  + The system should provide a live tracking feature that allows customers to monitor the real-time status and location of the mechanic once the service request is accepted and the mechanic is in route.
  + The tracking should include estimated arrival time and updates on the service progress.
* **FR6: Service History & Tracking**
  + The system should maintain a history of all service requests made by the customer, along with their statuses (e.g., Pending, In Progress, Completed).
  + Customers should be able to track the current status of active service requests in real time.
* **FR7: Payment Gateway Integration**
  + The system should integrate with secure payment gateways to enable customers to pay for services online using various payment methods (e.g., credit/debit cards, UPI, wallets).
  + The system should provide a receipt and confirmation of payment after successful transactions.
* **FR8: Notification System**
  + The system should send push notifications, SMS, or email alerts to customers for important events such as service request acceptance, mechanic arrival, and service completion.
* **FR9: Customer Support**
  + The system should include a customer support feature, such as a chat or contact form, to assist users with any issues or queries they might have.
* **FR10: Profile Management**
  + The system should allow customers to manage their profiles, including updating personal details, changing passwords, and viewing their service history.
* **FR11: Feedback and Rating**
  + After a service is completed, the system should prompt customers to provide feedback and rate the service provided by the mechanic.

**4.2. Mechanic Mobile and Web App**

* **FR12: User Registration & Login**
  + The system should allow mechanics to sign up and log in using their email or phone number.
  + The system should verify the mechanic's identity and credentials during registration.
* **FR13: Profile Management**
  + Mechanics should be able to manage their profiles, including updating personal details, setting availability, and adding service expertise.
* **FR14: Service Request Management**
  + The system should allow mechanics to view and manage incoming service requests (tokens), with options to accept, decline, or reschedule.
  + Mechanics should be able to view detailed information about the service request, including the customer’s contact details, product issue description, and location.
* **FR15: Service Execution Tracking**
  + Mechanics should be able to update the status of a service request as they progress through various stages (e.g., On the Way, In Progress, Completed).
  + The system should provide a live location-sharing feature, allowing customers to track the mechanic’s real-time location.
* **FR16: Earnings Dashboard**
  + The system should provide mechanics with a dashboard that displays their earnings, including a breakdown of completed services, payments received, and pending payments.
  + The dashboard should also show weekly and monthly earnings statistics.
* **FR17: Notification System**
  + The system should send push notifications, SMS, or email alerts to mechanics for new service requests, updates to existing requests, and payment confirmations.
* **FR18: Job History**
  + Mechanics should be able to view a history of all the jobs they have completed, including details such as date, service type, customer feedback, and earnings.
* **FR19: Availability Management**
  + The system should allow mechanics to set and update their availability, indicating when they are available to accept new service requests.
* **FR20: Feedback and Rating**
  + Mechanics should be able to view feedback and ratings provided by customers for completed services.
  + The system should allow mechanics to respond to customer feedback if necessary.
* **FR21: Customer Communication**
  + The system should provide mechanics with the option to contact customers directly via phone or in-app messaging to discuss service details or clarifications.

4.3**. Mechanic Mobile and Web App - Security Features**

* **FR22: Security & Verification**
  + The system should implement a mechanic verification process that includes background checks and identity verification.
  + Mechanics should be required to carry and display a company-issued ID badge when visiting customer homes.
  + The system should include a feature for customers to verify the mechanic’s identity through the app before allowing them entry.
* **FR23: Check-in/Check-out Feature**
  + Mechanics should use the app to check in when they arrive at the customer’s location and check out when they leave. This information should be logged and accessible to both the mechanic and the customer.
  + The system should notify the customer when the mechanic checks in and checks out.
* **FR24: Incident Reporting**
  + The system should allow mechanics to report any incidents or issues encountered during service directly through the app.
  + These reports should be sent to the company’s support team for review and action.

4.4. **Mechanic Mobile and Web App - SOS Features**

* **FR25: SOS Functionality**
  + The system should include an SOS button that mechanics can press in case of emergencies during service. This button should trigger an alert to emergency contacts or authorities.
  + The SOS alert should include the mechanic’s current location and a pre-set emergency message.
* **FR26: Emergency Contact Integration**
  + Mechanics should be able to set up emergency contacts within the app, who will be notified automatically when the SOS button is pressed.
  + The system should allow mechanics to customize the emergency message that is sent when the SOS function is activated.

**4.5. Customer Mobile and Web App - SOS Features**

* **FR27: SOS Functionality**
  + The system should include an SOS button that customers can press in case of emergencies during a service. This button should trigger an alert to emergency contacts or authorities.
  + The SOS alert should include the customer’s current location and a pre-set emergency message.
* **FR28: Emergency Contact Integration**
  + Customers should be able to set up emergency contacts within the app, who will be notified automatically when the SOS button is pressed.
  + The system should allow customers to customize the emergency message that is sent when the SOS function is activated.
* **FR29: Mechanic Verification on Arrival**
  + The system should prompt customers to verify the mechanic’s identity upon arrival by scanning a QR code or entering a unique code provided by the mechanic.
  + The system should alert the customer if the verification fails, with an option to trigger the SOS function immediately.

**5. Non-Functional Requirements**

* **Performance:** The application should be responsive and load within 3 seconds.
* **Scalability:** The system should support up to 100,000 concurrent users.
* **Security:** The application must adhere to standard security practices, including data encryption and secure login protocols.
* **Usability:** The app interface should be intuitive and easy to navigate for both customers and mechanics.
* **Availability:** The application should have an uptime of 99.9%.

**6. Assumptions**

* Customers and mechanics have access to a stable internet connection.
* The system will rely on third-party payment gateways.
* Mechanics will be equipped with the necessary tools and skills to complete the requested services.

**7. Dependencies**

* Integration with a secure payment gateway.
* Push notification services.
* Hosting services for web and mobile applications.

**8. Timeline**

* **Phase 1:** Requirement gathering and analysis - [Start Date] to [End Date]
* **Phase 2:** Design and prototyping - [Start Date] to [End Date]
* **Phase 3:** Development - [Start Date] to [End Date]
* **Phase 4:** Testing - [Start Date] to [End Date]
* **Phase 5:** Deployment - [Start Date] to [End Date]

**9. Budget**

* [Include estimated budget or cost analysis if required.]

**10. Stakeholders**

* **Client:** [Client Name]
* **Development Team:** [Team Name or Roles]
* **End Users:** Customers and Mechanics

**11. Approval**

This document has been reviewed and approved by the client and the development team.

* **Client Signature:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* **Date:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_